UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/611,688	06/30/2003	Matthew E. Miller	MSI-1615US	5312

22801   7590   01/17/2007

LEE & HAYES PLLC
421 W RIVERSIDE AVENUE SUITE 500
SPOKANE, WA 99201

EXAMINER
WEI, ZHENG

ART UNIT	PAPER NUMBER
2192	

SHORTENED STATUTORY PERIOD OF RESPONSE	NOTIFICATION DATE	DELIVERY MODE
3 MONTHS	01/17/2007	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Notice of this Office communication was sent electronically on the above-indicated "Notification Date" and has a shortened statutory period for reply of 3 MONTHS from 01/17/2007.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

lhptoms@leehayes.com

PTOL-90A (Rev. 10/06)

	Application No.	Applicant(s)
Office Action Summary	10/611,688	MILLER ET AL.
	Examiner	Art Unit	
	Zheng Wei	2192	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on <u>30 June 2006</u>.

2a)☐ This action is **FINAL**.   2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-42* is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-42* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on <u>30 June 2003</u> is/are: a)☒ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All b)☐ Some * c)☐ None of:

      1.☐ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *2/6/2004, 4/15/2004*.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application

6) ☐ Other: _____ .

DETAILED ACTION

1. This office action is in response to the application filed on 06/30/2003.

2. Claims 1-42 are pending and have been examined.

Oath/Declaration

3. The Office acknowledges receipt of a properly signed oath/declaration filed on February 18, 2004.

Priority

4. The priority date considered for this application is June 30, 2003.

Information Disclosure Statement

5. The information disclosure statements filed 02/06/2004 and 04/15/2004 have been placed in the application file and the information referred to therein has been considered.

Drawings

6. The drawings filed on June 30, 2003 are accepted by the Examiner.

Claim Rejections - 35 USC § 101

7.   35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

8.   Claims 5, 13, 15, 28, 36 and 39-42 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

Claims 5, 13, 28 and 36:

Claims 5, 13, 28 and 36 recite "A computer readable medium having data structures..." as the claimed subject matter. However, the data structure associated with the computer readable medium does not impart any functionality when employed as a computer element. Therefore they do not produce a useful, concrete, tangible result and are thus not statutory.

Claims 15 and 39:

Claims 15 and 39 recite "A computer readable medium having stored thereon a data structures..." as the claimed subject matter. However, the data structure associated with the computer readable medium does not impart any functionality when employed as a computer element. Therefore they do not produce a useful, concrete, tangible result and are thus not statutory

Claim 40:

Claim 40 recites "A computer readable medium having stored thereon information arranged according to schema data structures..." as the claimed

subject matter. However, the information and schema data structures associated

with the computer readable medium does not impart any functionality when

employed as a computer element. The information only comprises data or

software listings Per Se. Therefore it does not produce a useful, concrete,

tangible result and is thus not statutory.

Claims 41 and 42:

Claims 41 and 42 are dependent claims of claim 40. There claims all fail to

remedy the 35 U.S.C. 101 nonstatutory problem of claim 40. Therefore, they are

also rejected accordingly.

Claim Rejections - 35 USC § 102

9.   The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that

form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

10.   Claims 24-28 are rejected under 35 U.S.C. 102(b) as being anticipated by

Vaughan (Vaughan et al., GNU Autoconf, Automake and Libtool, 1st edition,

Published on Oct. 2000)

Claim 24:

Vaughan discloses an apparatus for generating configuration instructions used to

build a programmable machine, comprising:

- a library having:

    - a plurality of objects representing aspects of a configuration process

      (see for example, p.23, line 6- p.24, line 8, "aclocal", "autoconf",

      "automake" and related macro); and

    - a plurality of parameters associated with respective objects, wherein at

      least one of the parameters includes an unspecified value (see for

      example, p.21, lines 23-24, user input "configure.in", "makefile.am" as

      parameters for "autoconf" and "automake" discussed above); and

- build management logic configured to specify a set of objects from the

  library to implement the configuration of the programmable machine, and

  to generate the configuration instructions from the set of objects (see for

  example, p.22, lines 6-10, "The 'configure.in' is a template of macro

  invocations and shell code fragments used by autoconf to produce a

  'configure' script"), wherein the build management logic is further

  configured to generate the configuration instructions by determining the

  value of the unspecified value (see for example, p.35, lines 6-11, macro

code example to determine if the particular target system is sparc machine

or mips processor, then do different something, e.g., "sparc* -sun-solaris*)

do something ;;"; "mips*-*-elf*) do something ;;")

Claim 25:

Vaughan discloses the method according to claim 24, wherein the set of objects

from the library have a hierarchical order, and wherein the at least one parameter

that includes the unspecified value is associated with an object located at a

defined level within the hierarchal order (see for example, p.23, lines 8-10,

"Because 'configure.in' contains macro vocations that not known to Autoconf

itself –AM_INIT_AUTOMAKE being a case in point—it is necessary to collect all

the macro definitions for Autoconf to use when generating 'configure'.").

Claim 26:

Vaughan also discloses the method according to claim 25, wherein the build

management logic is configured to determine the unspecified value by

determining the value from an object that is higher in the hierarchical order than

the defined level ((see for example, p.38, lines 10-21, "The filename to include

can start with '$(top_srcdir)' to indicate that it should be found relative to the

topmost directory of the project; if it is a relative path or if it stars with '$(srcdir)', it

is relative to the current directory.").

Claim 27:

Vaughan further discloses the method according to claim 24, wherein the at least one parameter that includes the unspecified value comprises an expression that identifies a location to determine the value, and wherein the build management logic is configured to specify the value by accessing the location specified in the expression (see for example, p.38, lines 10-21, "Automake supports include directives", e.g., "include $(top_scrdir)/config/Make-rules" to indicate the location of the make rules).

Claim 28:

Vaughan discloses a computer readable medium having data structures and machine readable instructions for implementing the library and the build management logic of claim 24 (see for example of running the "automake", p.39, lines 15-16, "$ automake" and output message "automake: Makefile.am: not supported: source file 'subdir1/something.c' is in subdirectory" indicates that automake and Makefile.am are stored in the computer readable medium and execute by the computer to implement the library and the build management logic discussed as in claim 24 above.)

11.   Claims 29 and 31-36 are rejected under 35 U.S.C. 102(b) as being anticipated by

DuBois (Paul DuBois, "Software Portability with imake", 1st edition, Published on

July 1993)

Claim 29:

DuBois discloses a method for generating configuration instructions used to build

a programmable machine, comprising:

- a library having generic objects representing aspects of a configuration

  process (see for example, p.23, Figure 2-1, elements "Imakefile",

  "configuration files" and related text); and

- build management logic configured to generate the configuration

  instructions used to build the programmable machine by organizing the

  generic objects in the library based on a framework established by a

  template (see for example, p.23, Figure 2-1, elements "Imakefile",

  "configuration files", "Makefile" and related text, also see p.64, Figure 4-2

  about template file, "Section of Imake.tmpl specifying configuration

  defaults").

Claim 31:

DuBois discloses the method according to claim 29, wherein the build

management logic is configured to transfer the template to another user, or

receive the template from the other user (see for example, p.10, Figure 1-4, "Software development and porting with imake and make", steps about transferring software and Imakefile from Machines 1 to Machines 3 and related text description).

Claim 32:

DuBois also discloses the method according to claim 29, wherein the build management logic is configured to transfer the template to a head-end site, or receive the template from the head-end site(Machine 2 or Machine 3) (see for example, p.10, Figure 1-4, "Software development and porting with imake and make", steps about transferring software and Imakefile from Machines 1 to Machines 3 and related text description).

Claim 33:

DuBois further discloses the method according to claim 29, wherein the build management logic is configured to encapsulate information obtained from the library and the template in a package, and to transfer the package to another site (see for example, p.10, Figure 1-4, "Software development and porting with imake and make", steps 1-3, "Machine 1: write software and Imakefile", "move software to Machine 2" and "Machine 2: run imake to generate Makefile build software using Makefile" and related descriptions).

Claim 34:

DuBois also discloses the method according to claim 29, wherein the build

management logic is configured to generate a plurality of sets of configuration

instructions to build a respective plurality of programmable machines (see for

example, p.23, Figure 2-1, generates Makefile by using input files "Imakefile and

configuration files" and also see p.24, lines 11-27, three generated Makefiles with

a plurality of sets of configuration instructions to build three different systems)

Claim 35:

DuBois further discloses an method according to claim 34, wherein the build

management logic is configured to generate a synchronization file (rule macro)

that specifies a manner in which the configuration of each machine in the plurality

of programmable machines impacts other machines within the plurality of

programmable machines (see for example, p.35-36, "Recognize Target-building

Patterns" and p.38 example of rule macro)

Claim 36:

DuBois also disclose a computer readable medium having data structures and

machine readable instructions for implementing the method of claim 29 (see for

example, p.10, Figure 1-4, steps 1-3, "move software to Machine 2" and "move

software to Machine 3" to implement the method as discussed in claim 29 above)

12.   Claims 14 , 16-23 and 37-39 are rejected under 35 U.S.C. 102(e) as being

anticipated by <u>Suorsa</u> (Suorsa et al., US 7,152,109 B2)

Claim 14:

<u>Suorsa</u> discloses a system for generating configuration (provision) instructions

used to build a programmable machine, comprising:

- a head-end site (see for example, Fig.7, element 31 and related text),

  including:

  - head-end logic configured to interact with a remote client site (see for

    example, Fig.7, element 38, "communication Gateway" and related text);

    and

  - a central database coupled to the head-end logic, the central database

    containing at least one package that specifies configuration instructions

    (see for example, Fig.7, element 32, "Central Database" and related text),

    said at least one package including:

    - a plurality of objects representing aspects of a configuration

      process (see for example, col.9, lines 54-55, "This database

      comprises a repository of all pertinent information about each of

      the devices");

- a plurality of parameters associated with respective objects (see for example, col.9, lines 60-61, "The information stored in this database comprises all data that is necessary to provision a device"); and

- at least one template(model) for organizing the plurality of objects in accordance with a predetermined framework (see for example, col.15, lines 14-16, "the model for the intended configuration"); and

- a configuration site (see for example, Fig.7, devices 1...N and related text), including:

  - a local database for storing configuration instructions used to configure at least one machine associated with the configuration site (see for example, Fig.7, element 36, "agent" and related text); and

  - logic configured to receive and store said at least one package in the local database (see for example, Fig.10 and also see col.10, lines 39-40, "the agent 36 communicate with the central file system... to retrieve the required packages")

  - logic configured to generate configuration instructions used to configure at least one programmable machine based on said at least one package (see for example, col.10, lines 44-46, "commands can also be sent to the agent to instruct it to remove certain software, to configure the network

portion of the operating system...").

Claim 16:

Suorsa discloses an apparatus for generating configuration instructions used to build a programmable machine, comprising:

- a library having:

    - a plurality of generic objects representing aspects of a configuration process (see for example, col.9, lines 54-55, "This database comprises a repository of all pertinent information about each of the devices"); and

    - a plurality of parameters associated with respective generic objects (see for example, col.9, lines 60-61, "The information stored in this database comprises all data that is necessary to provision a device");

- build management logic configured to specify a set of objects from the library to implement the configuration of the programmable machine, and configured to generate the configuration instructions from the set of objects (see for example, col.10, lines 44-46, "commands can also be sent to the agent to instruct it to remove certain software, to configure the network portion of the operating system..."); and

- a user interface configured to allow a user to interact with the build management logic (see for example, Fig.7, element 40, "User Interface" and related text).

Claim 17:

Suorsa further discloses the apparatus according to claim 16, wherein the user

interface further includes a tree display section configured to display objects

organized as a hierarchical tree (see for example, Fig.9, "Rol XYZ" and related

text).

Claim 18:

Suorsa discloses the apparatus according to claim 17, and further discloses

wherein the user interface further includes a parameter display section

configured to display information pertaining to parameters that are associated

with at least one of the objects in the tree display section (see for example,

col.12, lines 29-31, "Rather, through the user interface, the operator first modifies

the model for that device which stored in the database").

Claim 19:

The apparatus according to claim 18, wherein the user interface further includes

a properties display section configured to display properties of at least one of the

objects in the tree display section or at least one parameter in the parameter

display section (see for example, col.12, lines 29-31, "Rather, through the user

interface, the operator first modifies the model for that device which stored in the

database").

Claim 20:

Suorsa also discloses the apparatus according to claim 16, wherein the build management logic includes logic configured to display ownership information at user interface. (see for example, col.17, lines 34-35, "Once the customer tier identification number has been determined"),

Claim 21:

Suorsa further discloses the apparatus according to claim 16, wherein the build management logic includes logic configured to display version information associated with information stored in the library (see for example, col.12, lines 34-41, "Preferably, the version history of the module is stored as well").

Claim 22:

Suorsa further discloses the apparatus according claim 16, wherein the build management logic further includes logic configured to restrict a user's right to manipulate information stored in the library based on the user's membership in one of a plurality of groups (see for example, col.13, lines 52-55, "The definition of the roles to be assigned to a device and stored in the database 32 is carried out through the user interface 40. The different roles can be associated with different access rights, to thereby affect their ability to be manipulated.")

Claim 23:

Suorsa also discloses the apparatus according to claim 16, wherein the build

management logic includes logic configured to apply validation rules to the entry

of parameter information to determine whether the entered parameter information

meets predetermined criteria (see for example, col.14, lines 37-39, "Thus, the

present invention provides a technique whereby the validity of a message or a

command transmitted to an agent may be verified").

Claim 37:

Suorsa discloses a method for processing requests for configuration instructions,

comprising:

- receiving a request from at least one configuration site for a configuration

    package, the configuration package including the configuration

    instructions(see for example, col.10, lines 38-40, "Upon receiving the address

    of the appropriate software, the agent 36 communicates with the central file

    system 34 to retrieve the required packages" also see col.9, lines 60-65, "The

    information stored in this database comprises all data that is necessary to

    provision a device");

- accessing a central database to retrieve the requested configuration package

    (see for example, p.10, lines 38-40, "Upon receiving the address of the

    appropriate software, the agent 36 communicates with the central file system

    34 to retrieve the required packages"); and

- transmitting the requested configuration package to the configuration site for
  its use in configuring at least one machine at the configuration site (see for
  example, Fig.5, steps 2a and 3a "configure" and related text; also see col.10,
  lines 42-44, "The commands that are sent to the agent also instruct it to
  configure..."),

wherein the configuration package include:

- a plurality of objects representing aspects of a configuration process (see for
  example, col.9, lines 54-55, "This database comprises a repository of all
  pertinent information about each of the devices");

- a plurality of parameters associated with respective objects (see for example,
  col.9, lines 60-61, "The information stored in this database comprises all data
  that is necessary to provision a device"); and

- at least one template for organizing the plurality of objects in accordance with
  a predetermined framework (see for example, col.15, lines 14-16, "the model
  for the intended configuration").

Claim 38:

Suorsa also discloses an apparatus for generating configuration instructions
used to build a programmable machine, comprising:

- a database having:

- a plurality of configuration items representing aspects of a configuration

  process (see for example, col.9, lines 54-55, "This database comprises a

  repository of all pertinent information about each of the devices"); and

- a plurality of features associated with respective configuration items,

  wherein at least one of the configuration items includes at least one of the

  following features: parameter information pertaining to at least one

  parameter associated with the configuration item; ownership information

  identifying an individual assigned ownership of the configuration item; and

  validation information identifying at least one validation rule applicable to

  the configuration item (see for example, col.11, lines 9-11, "Before

  releasing the package to the agent, the file server can check with the

  central database to determine whether the agent should have access

  rights to that package"); and

- build management logic configured to specify a set of configuration items

  from the database to implement the configuration of the programmable

  machine, and configured to generate the configuration instructions from the

  set of items (see for example, col.10, lines 42-44, "The commands that are

  sent to the agent also instruct it to configure...").

Claim 39:

Claim 39 claims a computer readable medium having stored thereon a data

structure, which recites the same limitation of the method claim and apparatus in

claim 37 and 38 respectively, wherein all claimed limitations have been address

and/or set forth above. Therefore, as the reference teaches all the limitation of

claim 37 or 38, it also anticipates the claim 39.

Claim Rejections - 35 USC § 103

13.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set
forth in section 102 of this title, if the differences between the subject matter sought to be patented and
the prior art are such that the subject matter as a whole would have been obvious at the time the
invention was made to a person having ordinary skill in the art to which said subject matter pertains.
Patentability shall not be negatived by the manner in which the invention was made.

14.   Claims 7, 30 and 42 are rejected under 35 U.S.C. 103(a) as being unpatentable

over DuBois (Paul DuBois, "Software Portability with imake", 1st edition,

Published on July 1993) in view of Jacquemot (Jacquemot et al., US

2004/0003388)

Claim 30:

DuBois discloses the method according to claim 29, wherein the template

(project.tmpl or Imake.tmpl) is expressed in text file format (see for example,

p.64, Figure 4-2, example of Imake.tmpl file), but does explicitly disclose that the

template is expressed in a markup language and has a form defined by a

schema. However, Jacquemot in the same analogous art of preparation of a

software configuration discloses using an XML type programming language (see

for example, p.1, paragraphs[0007], [0016] and [0022]). Therefore, it would have

been obvious to one having ordinary skill in the art at the time the invention was made to use XML to write template files in <u>DuBois</u>' teaching. One has motivation to use XML, because it is simple, standard and flexible in use as suggested by <u>Jacquemot</u> (see for example, p.1, paragraph [0007], "a form which is simple, standard an flexible in use")

Claim 7:

Claim 7 is a product version, which recites the same limitation of the method claim in claim 30, wherein all claimed limitations have been address and/or set forth above. Therefore, as the reference teaches all the limitation of claim 30, it also teaches the limitations of claim 7.

Claim 42:

<u>DuBois</u> the computer readable medium according to claim 40, wherein the schema is a plain text format (makefile), but does not explicitly disclose the schema is a markup language schema. However, <u>Jacquemot</u> in the same analogous art of preparation of a software configuration discloses using an XML type programming language (see for example, p.1, paragraphs[0007], [0016] and [0022]). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to use XML to write template files in <u>DuBois</u>' teaching. One has motivation to use XML, because it is simple, standard

and flexible in use as suggested by <u>Jacquemot</u> (see for example, p.1, paragraph
[0007], "a form which is simple, standard an flexible in use")

15.   Claims 1-5 are rejected under 35 U.S.C. 103(a) as being unpatentable over
<u>Vaughan</u> (Vaughan et al., GNU Autoconf, Automake and Libtool, 1st edition,
Published on Oct. 2000)

Claims 1-5:

Claims 1-5 are a product version, which recite the same limitations of those
method claims in claims 24-28, wherein all claimed limitations have been
address and/or set forth above. Therefore, as the references teach all the
limitation of claims 24-28, they also teach the limitations of claims 1-5
respectively.

16.   Claims 6, 8-13, 15 and 40-42 are rejected under 35 U.S.C. 103(a) as being
unpatentable over <u>DuBois</u> (Paul DuBois, "Software Portability with imake", 1st
edition, Published on July 1993)

Claims 6 and 8-13:

Claims 6 and 8-13 are a product version, which recite the same limitations of
those method claims in claims 29 and 31-36, wherein all claimed limitations have
been address and/or set forth above. Therefore, as the references teach all the

limitations of claims 29 and 31-36, they also teach the limitations of claims 6 and 8-13 respectively.

Claim 15:

Claims 15 is another product version, which recite the same limitations of the method claim in claim 29, wherein all claimed limitations have been address and/or set forth above. Therefore, as the references teach all the limitations of claim 29, it also teaches the limitations of claim 15.

Claim 40:

DuBois discloses a method to configure and make software build for different types of computer system, the method discloses:

- a machine element pertaining to a machine to be configured (see for example, p.23, Figure 2-1, "configuration files: complex and contain machine-specifics" and related text);

- a stage element pertaining to a stage involved in configuring the machine (see for example, p.46, generated makefile example, stage element: "proga" and "install");

- a phase list element pertaining to a list of phases used to implement the stage (see for example, p.46, generated makefile example, phase list: "cc –o proga proga.o –lm");

- a parameter element pertaining to a parameter associated with the configuration of the machine (see for example, p.46, generated makefile example, parameter element: "$(CC)"); and

- a value element pertaining to a value assigned to the parameter (see for example, p.46, generated makefile example, value element: "CC=cc").

Claim 40 claims a computer readable medium, which is a product version recites the same limitations of the method discussed above, wherein all claimed limitations have been address and/or set forth. Therefore, as the references teach all the limitations of above, claim 40 is unpatentable.

Claim 41.

DuBois also discloses the computer readable medium according to claim 40, further including: a group element referring to group in which the machine is a member (see for example, p.46, generated makefile example, "#include INCLUDE_IMAKEFILE" and comments).

Conclusion

17.   The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

18.   Any inquiry concerning this communication or earlier communications from the examiner should be directed to Zheng Wei whose telephone number is (571)
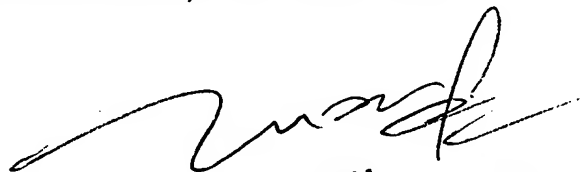
270-1059. The examiner can normally be reached on Monday-Thursday 8:00-

15:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's

supervisor, Tuan Q. Dam can be reached on (571) 272-3695. The fax phone number

for the organization where this application or proceeding is assigned is 571-273-8300.

Any inquiry of a general nature of relating to the status of this application or

proceeding should be directed to the TC 2100 Group receptionist whose telephone

number is 571- 272-1000.

Information regarding the status of an application may be obtained from the

Patent Application Information Retrieval (PAIR) system. Status information for

published applications may be obtained from either Private PAIR or Public PAIR.

.Status information for unpublished applications is available through Private PAIR only.

For more information about the PAIR system, see http://pair-direct.uspto.gov. Should

you have questions on access to the Private PAIR system, contact the Electronic

Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a

USPTO Customer Service Representative or access to the automated information

system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

ZW

TUAN DAM
SUPERVISORY PATENT EXAMINER